WORKING PAPER
ALFRED P. SLOAN SCHOOL OF MANAGEMENT

A STOCHASTIC APPROACH TO GLOBAL OPTIMIZATION

by A.H.G. Rinnooy Kan * ***
C.G.E. Boender * **
G.Th. Timmer * **

WP1602-84 October 1984

MASSACHUSETTS
INSTITUTE OF TECHNOLOGY
50 MEMORIAL DRIVE
CAMBRIDGE, MASSACHUSETTS 02139

A STOCHASTIC APPROACH TO GLOBAL OPTIMIZATION

by A.H.G. Rinnooy Kan * ***

C.G.E. Boender * **

G.Th. Timmer * **

WP1602-84 October 1984

* Econometric Institute, Erasmus University Rotterdam, The Netherlands
** Department of Mathematics and Informatics, Delft University of Technology, The Netherlands
*** Sloan School of Management, M.I.T., Cambridge, Massachusetts

1. INTRODUCTION

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a real valued smooth *objective function*. The area of *nonlinear programming* is traditionally concerned with methods that find a *local optimum* (say local minimum) of f, i.e. a point $x^* \in \mathbb{R}^n$ such that there exists a neighbourhood B of x^* with

$$f(x^*) \leq f(x) \qquad \forall\, x \in B. \tag{1}$$

In general, however, *several* local optima may exist and the corresponding function values may differ substantially. The *global optimization problem* is to find the *global optimum* (say global minimum) x_* of f, i.e. to find a point $x_* \in \mathbb{R}^n$ such that

$$f(x_*) \leq f(x) \qquad \forall\, x \in \mathbb{R}^n. \tag{2}$$

For computational reasons one usually assumes that a convex and compact set $S \subset \mathbb{R}^n$ is specified in advance, which contains the global minimum as an interior point. None the less, the problem to find

$$y_* = f(x_*) = \min_{x \in S} f(x) \tag{3}$$

remains essentially one of *unconstrained* optimization, and as such forms the subject of this paper.

So far only few solution methods for the global optimization problem have been developed, certainly in comparison with the multitude of methods that aim for a local optimum. The relative difficulty of global optimization as compared to local optimization is easy to understand. It is well known that under the assumption that f is twice continuously differentiable, all that is required to test if a point is a local minimum is knowledge of the first and second order derivatives at this point. If the test does not yield a positive result, the smoothness properties of f ensure that a neighbouring point can be found with a lower function value. Thus, a sequence of points can be constructed that converges to a local minimum.

Such local tests are obviously not sufficient to verify global optimality. Indeed, the global optimization problem as stated in (3) is inherently unsolvable [Dixon 1978]: for any continuously differentiable function f, any point $\bar{x} \in S$ and any neighbourhood B of $\bar{x}$, there exists a function f' such that (i) f+f' is continuously differentiable, (ii) f+f' is equal to f in all points outside B, and (iii) the global minimum of f+f' is attained in $\bar{x}$. As B can be chosen arbitrarily small, it immediately follows that it requires an unbounded number of function evaluations to guarantee that the global minimum x_* will be found.

Of course, this argument does not apply when one is satisfied with an approximation of the global minimum. In particular, for the case that a point within distance ε from x_* is sought, enumerative strategies exist that only require a finite number of function evaluations. These strategies, however, are of limited practical use. Thus, either a further restriction of the class of objective functions or a further relaxation of what is required of an algorithm will be inevitable in what follows.

Subject to this first conclusion the methods developed to solve the global optimization problem can be divided in deterministic and stochastic methods.

Some deterministic methods will be reviewed in Section 2. If a rigid guarantee is desired for these methods, the previous argument indicates that additional assumptions about f are unavoidable. The most popular such assumption is that a Lipschitz constant L is given, i.e. for all x_1, $x_2 \in S$

$$|f(x_1) - f(x_2)| \leq L\|x_1-x_2\|, \qquad (4)$$

where $\|.\|$ denotes the Euclidean distance. The upper bound on the rate of change of f implied by this Lipschitz constant can be used in various ways to perform an exhaustive search over S. In practice, however, it is impossible to verify whether a function satisfies such a Lipschitz condition or not. In addition, the computational effort required by these methods tends to be formidable and forbidding.

Better computational results are obtained by methods that exploit the continuous differentiability of f. As mentioned before, this property allows for the construction of a sequence of points converging to a local optimum. As there exists no local test to verify global optimality, these deterministic methods try to find the global minimum by locating all local minima. No such method, however, can truly guarantee that all local minima of f are really found. Thus, as we shall see, their superior computational results are obtained at the expense of more (possibly implicit) assumptions about f or of no certainty of success.

Generally, far better results - both theoretically and computationally - have been obtained by stochastic methods [Rinnooy Kan & Timmer 1984, Timmer 1984]. In most stochastic methods, two phases can be usefully distinguished. In the global phase, the function is evaluated in a number of randomly sampled points. In the local phase , the sample points are manipulated, e.g. by means of local searches, to yield a candidate global minimum.

Generally in turning to stochastic methods, we do sacrifice the possibility of an absolute guarantee of success. However, under mild conditions on the sampling distribution and on f, the probability that a feasible solution within distance ε of x_* is sampled will be seen to approach 1 as the sample size increases [Solis & Wets 1981]. If the sample points are drawn from a uniform distribution over S and if f is continuous, then an even stronger result holds: the sample point with lowest function value converges to the global minimum value with probability 1 (or almost surely). Thus, the global phase can yield an asymptotic guarantee with probability 1, and is therefore essential for the reliability of the method. However, a method that only contains a global phase will be found lacking in efficiency. To increase the latter while maintaining the former is one of the challenges in global optimization.

Stochastic methods will be discussed in Section 3. The most promising methods appear to be variants of the so-called Multistart technique where points are sampled iteratively from a uniform distribution over S (global phase), after which local minima are found by applying a local search procedure to these points (local phase).

In practice, the number of local minima of an objective function is usually unknown. A fortiori, it is uncertain if a sample of observed local minima includes the global one. Thus, in this approach there is typically a need for a proper stopping rule. A theoretical framework which provides a solution to this problem is developed in [Boender 1984]. It turns out to be possible, for example, to compute a Bayesian estimate of the number of local minima not yet identified, so that the sequence of sampling and searching can be stopped if the estimated number of local minima is equal to the number of minima identified.

Multistart is still lacking in efficiency because the same local minimum may be located several times. If we define the region of attraction R_{x*} of a local minimum x* to be the set of points in S starting from which a given local search procedure converges to x*, then ideally, the local search procedure should be started exactly once in every region of attraction. Several new algorithms designed to satisfy this criterion are presented in [Timmer 1984].

The method discussed in Section 3.3 temporarily eliminates a prespecified fraction of the sample points whose function values are relatively high. The resulting reduced sample consists of groups of mutually relatively close points that correspond to the regions with relatively small function values. Within each group the points are still distributed according to the original uniform distribution. Thus, these groups can be identified by clustering techniques based upon tests on the uniform distribution. Only one local search procedure will be started in each group [Boender et al. 1982].

Unfortunately, the resulting groups do not necessarily correspond to the regions of attraction of f. It is possible that a certain group of points corresponds to a region with relatively small function values which contains several minima. Therefore, the method which is based on the reduced sample may fail to find a local minimum although a point is sampled in its region of attraction. A better method is described in Section 3.4. Here, the function value is used explicitly in the clustering process. A very simple method results, for which both the probability that the local search procedure is started unnecessarily, and the probability that the local search is not started although a new local minimum would have been found, approach 0 with increasing sample size. In some sense the results proven for this method can be seen to be the strongest possible ones.

The results of some computational experiments are reported in Section 4.

2. DETERMINISTIC METHODS

2.1. Finite exact methods

We first consider exact methods that provide an absolute guarantee that the global minimum will be found in a finite number of steps.

Space covering methods exploit the availability of a Lipschitz constant L (cf. (4)) to perform an exhaustive search over S. A conceptually simple method of this type has been proposed by Evtushenko [Evtushenko 1971]. Suppose that f has been evaluated in $x_1,\ldots,x_k$ and define $M_k = \min\{f(x_1),\ldots,f(x_k)\}$. If the spheres V_i $(i=1....,k)$ are chosen with centre x_i and radius $r_i = (f(x_i) - M_k + \varepsilon)/L$, then for any $x \in V_i$

$$f(x) \geq f(x_i) - Lr_i = M_k - \varepsilon. \qquad (5)$$

Hence, if the spheres V_i $(i=1,\ldots,k)$ cover the whole set S, M_k differs less than ε from y_*. Thus, this result converts the global minimization problem to the problem of covering S with spheres. In the simple case of 1-dimensional optimization where S is an interval $\{x \in \mathbb{R} | a \leq x \leq b\}$, this covering problem is solved by choosing $x_1 = a + \varepsilon/L$ and

$$x_k = x_{k-1} + \frac{2\varepsilon + f(x_k) - M_k}{L} \qquad k = 2,3,\ldots \qquad (6)$$

The method obviously stops if $x_k \geq b$.

A generalization for higher dimensional problems $(n>1)$ consists of covering S with hypercubes whose edgelength is $2r_i/\sqrt{n}$, i.e. cubes inscribed in the spheres V_i.

Note that the efficiency of the method depends on the value of M_k. Since the distances between the iteration points increase with decreasing M_k, it may be worthwhile to improve M_k using a local minimization procedure.

A different method, for which it is not necessary to specify any a priori accuracy ε, is proposed in [Shubert 1972]. Here a bound on the accuracy is calculated at each iteration. The method consists of iteratively updating a piecewise linear function, which has directional derivaties equal to L or -L everywhere and which forms a lower bound on f that improves with each iteration. The method was orignally designed for 1-dimensional problems, but can be generalized to higher dimensional problems.

Initially, f is evaluated at some arbitrary point x_1. A piecewise linear function $\psi_1(x)$ is defined by

$$\psi_1(x) = f(x_1) - L\|x - x_1\|. \qquad (7)$$

Now an iterative procedure starts, where in iteration k ($k \geq 2$) a global minimum of $\psi_{k-1}(x)$ on S is chosen as the point where f is next evaluated. A new piecewise linear function $\psi_k(x)$ is constructed by a modification of $\psi_{k-1}(x)$.

$$\psi_k(x) = \max\{f(x) - L\|x - x_k\|, \psi_{k-1}(x)\} \qquad (k = 2,3,\ldots) \quad (8)$$

Hence,

$$\psi_{k-1}(x) \leq \psi_k(x) \leq f(x), \qquad (9)$$

$$\psi_k(x_i) = f(x_i) \qquad (i = 1,\ldots,k). \qquad (10)$$

In each iteration, the piecewise linear approximation for f will improve. The method is stopped when the difference between the global minimum of $\psi_k(x)$, which is a lower bound on the global minimum of f, and the best function value found is small enough.

To conclude the description of this method, note that $\psi_k(x)$ is completely determined by the location and the value of its minima. If $\psi_k(x)$ is decribed in terms of these parameters it is no problem to find one of its global minima.

Although the space covering techniques are intuitively appealing they have two major drawbacks. Firstly, the number of function evaluations required by these methods tends to be formidable. To analyse this number, let S be a hypersphere with radius r, so that

$$m(S) = \frac{r^n \pi^{n/2}}{\Gamma(1 + \frac{n}{2})}, \qquad (11)$$

where Γ denotes the gamma function.

Furthermore, let c be the maximum of f over S and suppose that f has been evaluated in k points $x_1,\ldots,x_k$. The function value in a point x can only be known to be greater than the global minimum value y_* if the function has been evaluated in a point x_i within distance $(f(x_i) - y_*)/L$ of x. Hence, the hyperspheres with radii $(f(x_i) - y_*)/L$ centered at the points x_i, $i = 1,\ldots,k$, must cover S to be sure that the global minimum has been found. The joint volume of these k hypersphere is smaller than

$$\frac{k\left(\frac{c-y_\star}{L}\right)^n \pi^{n/2}}{\Gamma(1+\frac{n}{2})} . \tag{12}$$

Thus, for the k hyperspheres to cover S we require

$$k > \left(\frac{r}{c-y_\star}\right)^n L^n. \tag{13}$$

Unless the derivative of f in the direciton of the global minimum equals -L everywhere, L is greater than $\frac{r}{c-y*}$, and the computational effort required increases exponentially with n.

A second drawback of the space covering techniques is that the Lipschitz constant has to be known or estimated before starting the minimization. Overestimating L raises the cost considerably (cf. (13)), while underestimating L might lead to failure of the method. In most practical cases, however, obtaining a close estimate of L poses a problem comparable in difficulty with the original global optimization problem. Both drawbacks seem inherent to the approach chosen.

Surprisingly good computational results have been obtained by a similar enumerative technique in which upper and lower bounds on f over a subset of S (say, a hypercube) are computed by interval arithmetic [Hansen 1980]. This approach presupposes that f is given as a (not too complicated) mathematical expression. This is the case for all the standard testproblems - though not always in practice - and on those problems the straightforward branch-and-bound procedure based on the above idea has performed very well indeed.

In addition to the enumerative methods mentioned above, an absolute guarantee of success can also be achieved for certain very special classes of functions, most notably polynomials.

If f is a one dimensional polynomial, then a deflation technique has been proposed by [Goldstein & Price 1971].

Consider the Taylor series around a local minimum x* of a one dimensional function f.

$$f(x) = f(x^*) + \frac{f^{(2)}(x^*)}{2!}(x-x^*)^2 + \frac{f^{(3)}(x^*)}{3!}(x-x^*)^3 + \dots + \frac{f^{(k)}(x^*+\theta(x-x^*))}{k!}(x-x^*)^k, \qquad (14)$$

where $0 \leq \theta \leq 1$ and $f^{(i)}(.)$ is the i-th order derivative of f. Now let

$$f_1(x) = \frac{f(x) - f(x^*)}{(x - x^*)^2} . \qquad (15)$$

If f is a polynomial of degree m, then $f_1(x)$ is a polynomial of degree m-2. If, in addition, it can be shown that the global minimum of $f_1(x)$ is positive, then x* is the global minimum of f. In case there is a point $\bar{x}$ for which $f_1(x)$ is negative, then $f(\bar{x}) < f(x^*)$ and x* is not the global minimum. In the latter case one can proceed using the Taylor series around a new local minimum which can be found by applying P to $\bar{x}$. To determine whether the global minimum is positive, we proceed iteratively considering $f_1(x)$ as the new basic function. If f(x) is a one dimensional polynomial, then this is a finite and rapidly converging process. For a more general function, however, there is no reason to assume that the problem of showing that the global minimum of $f_1(x)$ is positive is easier than the original problem.

Recently piecewise linear homotopy methods [Todd 1976, Allgower & Georg 1980] have proven to be useful in identifying all roots of polynomials, which is related to identifying all minima. Using a labeling rule it is possible to determine N points, such that all roots of a one dimensional polynomial of degree N will be found as the result of a simplicial path following algorithm applied to each of these points [Kuhn et al. 1984]. This can be implemented efficiently: it only takes $O(N^3 \log(N/\epsilon))$ evaluations of f to find a point which is within ϵ distance of a root of f. For details we refer to [Kuhn et al. 1984].

Polynomials are not the only class of functions for which methods have been proposed that exploit the specific features of that class. For instance, successively closer approximations of f, for which the global minimum can be easily calculated, can be determined if f is separable into convex and concave terms [Falk & Solund 1971, Solund 1971], if a convex envelope of f can be found [McCormick 1976], and if f can be written as a finite sum of products of a finite number of uniform continuous functions of a single argument [Beale & Forrest 1978].

2.2. Heuristic methods

We now turn to heuristic methods that only offer an empirical guarantee (i.e., they may fail to find the global optimum). These methods apply a local search procedure to different starting points to find the local minima of f.

The tunneling method attempts to solve the global optimization problem by performing local searches such that each time a different local minimum is reached [Levy & Gomez 1980].

The method consists of two phases. In the first phase (minimization phase) the local search procedure is applied to a given point x_0 in order to find a local minimum x^*. The purpose of the second phase (tunneling phase) is to find a point x different from x^*, but with the same function value as x^*, which is used as a starting point for the next minimization phase. This point is obtained by finding a zero of the tunneling function

$$T(x) = \frac{f(x) - f(x^*)}{\|x-x_m\|^{\lambda_0} \prod_{i=1}^{\ell} \|x-x_i^*\|^{\lambda_i}}, \qquad (16)$$

where $x_1^*,\ldots,x_\ell^*$ are all local minima with a function value equal to $f(x^*)$ found in previous iterations. Subtracting $f(x^*)$ from $f(x)$ eliminates all points satisfying $f(x) > f(x^*)$ as a possible solution. The term $\Pi_{i=1}^{\ell} \|x-x_i^*\|$ is introduced to prevent the algorithm from choosing the previously found minima as a solution. To prevent the zero finding algorithm to converge to a stationary point of

$$\frac{f(x) - f(x^*)}{\prod_{i=1}^{\ell} \|x-x_i^*\|^{\lambda_i}} \qquad (17)$$

which is not a zero of (16), the term $\|x-x_m\|^{\lambda_0}$ is added, with x_m chosen appropriately.

If the global minimum has been found, then (16) will become positive for all x. Therefore the method stops if no zero of (16) can be found.

The tunneling method has the advantage that, provided that the local search procedure is of the descent type, a local minimum with smaller function value is located in each iteration. Hence, it is likely that a point with small function value will be found relatively quickly. However, a major drawback of the method is that it is impossible to be certain that the search for the global minimum has been sufficiently thorough. In essence, the tunneling method only reformulates the problem: rather than solving the original minimization problem, one now must prove that the tunneling function does not have a zero. This, however, is once again a global problem which is strongly related to the original one. The information gained during the foregoing iterations is of no obvious use in solving this new global problem; which therefore appears to be as hard to solve as the original one. Thus, lacking any sort of guarantee, the method is at best of some heuristic value.

The same is true for the trajectory method due to Branin [Branin 1972, Branin & Hoo 1972], based on the construction (by numerical integration) of the path along which the gradient of f points in constant direction. This method is known to fail on certain functions [Treccani 1975], and it is not clear under which conditions convergence to a global minimum can be assured.

3. STOCHASTIC METHODS

Stochastic methods are asymptotically exact, i.e. they offer an a symptotic guarantee in some probabilistic sense. The methods can usefully be separated into two different phases.

In the global phase, the function is evaluated in a number of randomly sampled points. In the local phase, the sample points are manipulated, for example by means of local searches, to yield a candidate solution.

The global phase is necessary because there is no local improvement strategy which, starting from an arbitrary point, can be guaranteed to converge to the global minimum. As we have seen in Section 1, a global search over S, which in the long run locates a point in every subset of S of positive measure, is required to ensure the reliability of the method. But, although the local improvement techniques cannot guarantee that the global minimum will be found, they are efficient tools to find a point with relatively small function value. Therefore, the local phase is incorporated to improve the efficiency of the method. Because the local phase generally complicates the formal analysis considerably, we will start our survey with a method consisting only of a global phase.

3.1. Pure Random Search

The simplest stochastic method for global optimization consists only of a global phase. Known confusingly as Pure Random Search [Brooks 1958, Anderssen 1972], the method involves no more than a single step.

Pure Random Search

Step 1. Evaluate f in N points, drawn from a uniform distribution over S. The smallest function value found is the candidate solution for y_*.

The proof that Pure Random Search offers an asymptotic guarantee in a probabilistic sense is based on the observation that the probability that a uniform sample of size N contains at least one point in a subset $A \subset S$ is equal to [Brooks 1958]

$$1 - \left(1 - \frac{m(A)}{m(S)}\right)^N, \tag{18}$$

where m(.) denotes the Lebesgue measure. Thus Pure Random Search locates an element close to the global minimum with a probability approaching to 1 as N increases. In fact, if we let $\underline{y}_N^{(1)}$ be the smallest function value found in a sample of size N, then it can be proved that $\underline{y}_N^{(1)}$ converges to the global minimum value y_* with probability 1 [cf. Devroye 1978, Rubinstein 1981].

We also observe that (18) implies that

$$\frac{\log(1-\alpha)}{\log(1-\delta)} \tag{19}$$

sample points are required to find an element of a set A with probability α, provided that $m(A)/m(S) = \delta$. This result can be used to provide a stopping rule for this method in the obvious manner.

3.2. Multistart

In view of the extreme simplicity and the resulting poor computational quality of Pure Random Search, several extensions have been proposed that also start from a uniform sample over S (hence, the results of the foregoing section can be applied including the asymptotic guarantee), but that at the same time involve local searches from some or all points in the sample. In this section we will discuss the prototype of these methods which is known as Multistart. In this approach a local search procedure P is applied to each point in the random sample; the best local minimum found in this way is our candidate for the global minimum x_*.

Multistart

Step 1. Draw a point from the uniform distribution over S.

Step 2. Apply P to the new sample point.

Step 3. The local minimum x^* identified with the lowest function value is the candidate value for x_*. Return to Step 1, unless a stopping criterion is satisfied.

Let us consider the issue of a proper stopping criterion for this method. In the sequel we will show that the stopping rules developed for Multistart remain valid for more efficient variants of this folklore approach.

Recall that the region of attraction R_{x^*} of a local minimum x^*, given a particular local search routine P, is defined as the subset of points in S starting from which P will arrive at x^* [Dixon & Szegö 1975, 1978]. Furthermore, let k be the number of local minima of f, and denote the relative size of the i-th region of attraction by θ_i (i=1,...,k). If these values are given, we have several stopping criteria at our disposal. We may terminate the Multistart method, for example, if the number of different local minima observed is equal to k or if the total size of the observed regions of attraction is greater than some prespecified value.

In practice, $k,\theta_1,\ldots,\theta_k$ are frequently unknown. The sampled minima, however, clearly provide information about their values. The crucial observation that enables us to learn about the values of $k,\theta_1,\ldots,\theta_k$ is that since the starting points of the Multistart method are uniformly distributed over S, a local minimum has a fixed *probability* of being found in each trial that is equal to the relative size of its region of attraction. This implies that, given a number of local searches N, the observed local minima are a sample from a *multinomial distribution* whose *cells* correspond to the local minima: the *number of cells* is equal to the unknown number k of local minima of f and the *cell probabilities* are equal to the unknown relative sizes θ_i $(i=1,\ldots,k)$ of the regions of attraction. However, since it is unknown in what way S is subdivided in regions of attraction, it is impossible to distinguish between samples of local minima that are identical up to a relabeling of the minima. We therefore have to rely on the *generalized multinomial distribution* that has been studied in great detail in [Boender 1984]. It is now standard statistical practice to use an observed sample of local minima to make inferences about the unknown parameters $k,\theta_1,\ldots,\theta_k$. In a *Bayesian approach*, in which the unknowns are themselves assumed to be random variables with a *uniform prior distribution*, it can be proved that, given that W different local minima have been found in N searches, the *optimal Bayesian estimate* of the unknown number of local minima k is given by the integer E nearest to

$$W.\frac{N-1}{N-W-2} \quad (N \geq W-3). \tag{20}$$

(cf [Boender 1984]). Hence, the Multistart method can (for instance) be stopped when E = W.

This theoretical framework which was initiated in [Zielinski 1981] is an attractive one, the more so since it can easily be extended to yield *optimal Bayesian stopping rules* that incorporate assumptions about the costs and potential benefits of further local searches and weigh these against each other probabilistically. Several loss structures and corresponding stopping rules are described in [Boender 1984].

3.3. Single Linkage

In spite of the reliability of Multistart, the method is lacking in efficiency, which stems from the fact that each local minimum, particularly the ones with a large region of attraction, will generally be found several times. From efficiency considerations only, the local search procedure P should ideally be invoked no more than once in each region of attraction. Computationally successful adaptations of Multistart in that direction are provided by *clustering methods* [Becker & Lago 1970; Törn 1978; Boender et al. 1982; Timmer 1984]. Clustering methods also generate points iteratively in S according to the uniform distribution. Now, however, only a prespecified *fraction* q containing the points with the lowest function values are retained in the sample. Let f_q be the largest function value in the *reduced sample* and define $R_q \subset S$ as the set of all points in S whose function value does not exceed f_q. R_q will consist of a number of disjoint *components* that together contain all the points from the reduced sample: a nonempty set of all reduced sample points that are contained in one component of R_q is called a *cluster*. Ideally, the clusters should be in 1-1 correspondence with the regions of attraction whose intersection with R_q is nonempty. Then, one local search from the best point in each cluster will suffice to find the set of local minima with function value smaller than f_q, which obviously includes the *global* minimum.

In the *Single Linkage* global optimization algorithm [Timmer 1984], clusters are efficiently identified by exploiting the fact that the points in the reduced sample are uniformly distributed over R_q. Clusters are created one by one, and each cluster is initiated by a *seedpoint*. Selected points of the reduced sample are added to the cluster until a *termination criterion* is met. Under conditions to be specified, the local search procedure is started from one point in the cluster.

Before we state the algorithms we need some additional notation. Fix $\tau > 0$ and let S_τ denote the points in S whose distance to the boundary of S is at least τ. Furthermore, let X^* be the set of detected local minima, and given $\upsilon > 0$, let $X^*_\upsilon = \{x \in S \mid \|x-x^*\| < \upsilon$, for any $x^* \in X^*\}$. Henceforth it is assumed that (i) all local minima of f occur in the interior of S_τ, (ii) a positive constant ε can be specified such that the distance between any two stationary points of f exceeds ε, (iii) the local search procedure P always finds a local minimum x^*, and (iv) P is *strictly descent*, i.e. starting from any $x \in S$ P converges to a local minimum $x^* \in S$ such that there exists a path in S from x to x^* along which the function values are nonincreasing. We now describe the Single Linkage algorithm, given N uniform points in S.

Single Linkage

Step 1. (Determine reduced sample). Determine the reduced sample by taking qN sample points with the smallest function values. Let W be the number of elements of the set of local mninima X*. Set j := 1.

Step 2. (Determine seed points). If all reduced sample points have been assigned to a cluster, stop.
If $j \leq W$, then choose the j-th local minimum in X* as the next seedpoint; go to Step 3.
Determine the point $\bar{x}$ which has the smallest function value among the unclustered reduced sample points; $\bar{x}$ is the next seedpoint.
If $\bar{x} \in S_\tau$ and if $\bar{x} \in X^*_\upsilon$, then apply P to $\bar{x}$ to find a local minimum x*.

Step 3. (Form cluster). Initiate a cluster from a seedpoint which is determined in Step 2. Add reduced sample points which are within the critical distance r_N from a point already in the cluster until no more such points remain. Let j := j+1 and go to Step 2.

The sample is expanded and the above procedure repeated until the stopping rule applies.

Several observations are in order. First of all, in [Timmer 1984] it is proved that if the critical distance r_N is chosen equal to

$$\pi^{-\frac{1}{2}}\left(\Gamma(1+\tfrac{n}{2})m(S)\frac{\sigma\log N}{N}\right)^{1/n} \qquad (21)$$

with $\sigma > 2$ then the probability that a local search is started tends to 0 with increasing N; of $\sigma > 4$, then, even of the sampling continues forever, the total number of local searches ever started is finite with probability 1. In addition, whenever the critical distance tends to 0 for increasing N, then in very component in which a point has been sampled a local minimum will be found with probability 1.

Secondly, the stopping rules developed for Multistart can be applied to the clustering method provided that the number of trials is taken equal to the number of points qN in the reduced sample rather than the number of local searches, the number of local minima is taken equal to the number of local minima whose function value is not greater than f_q and the cell probabilities are taken to be equal to the relative Lebesgue measure of the intersections of the regions of attraction with R_q. In applying these rules, we do have to assume that the way R_q changes (slightly) with the sample size N does not affect the analysis. More importantly, we also have to assume that each local minimum with function value smaller than f_q whose region of attraction does contain at least one point from the reduced sample is actually found, i.e. that the methods identify the same local minima that would be found by performing a local search from each of the qN points in the reduced sample. This assumption is unfortunately not justified for the Single Linkage: a component way contain several local minima, of which we are only guaranteed to find one asymptotically.

3.4. Multi Level Single Linkage

The method described in Section 3.3 only makes minimal use of the function values of the sample points. These function values are used to determine the reduced sample, but the clustering process applied to this reduced sample hardly depends on the function values. Instead, the clustering process concentrates on the location of the reduced sample points. As a result, the method cannot distinguish between different regions of attraction which are located in the same component of R_q. The function value of a sample point x evidently can be of great importance if one wishes to predict to which region of attraction x belongs, because the local search procedure which defines these regions is known to be strictly descent. Hence, x cannot belong to the region of attraction of a local minimum x*, if there is no descent path from x to x*, i.e. a path along which the function values are monotonically decreasing. Furthermore, x does certainly belong to the region of attraction R_{x*}, if there does not exist a descent path from x to any other minimum than x*.

Obviously, it is impossible to consider all descent paths starting from x. Instead, we will (implicitly) consider all r_N-descent sequences, where a r_N-descent sequence is a sequence of sample points, such that each two successive points are within distance r_N of each other and such that the function values of the points in the sequence are monotonically decreasing. It will turn out that if the sample size increases and if r_N tends to 0, then every descent path can be conveniently approximated by such a sequence of sample points.

For a better understanding of the remainder of this section it is advantageous to consider the following algorithm first. Let W be the number of local minima known when the procedure is started.

<u>Step 1</u>. Initiate W different clusters, each consisting of one of the local minima present.

<u>Step 2</u>. Order the sample points, such that $f(x_i) < f(x_{i+1})$, $1 \leq i \leq N-1$. Set $i := 1$.

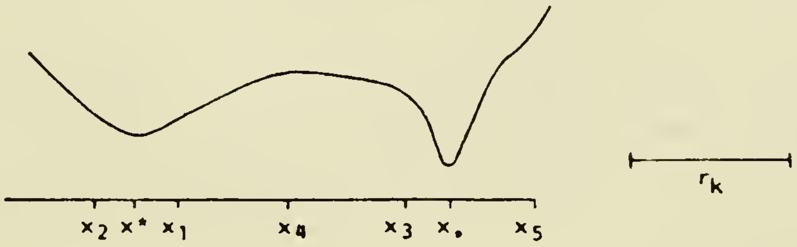
<u>Step 3</u>. Assign the sample point x_i to every cluster which contains a point within distance r_N.
If x_i is not assigned to any cluster yet, then start a local search at x_i to yield a local minimum x^*. If $x^* \notin X^*$, then add x^* to X^*, set $W := W+1$ and initiate the W-th cluster by x^*. Assign x_i to the cluster that is initiated by x^*.

<u>Step 4</u>. If $i = N$, then stop. Else, set $i := i+1$ and go to Step 3.

Note that a sample point x can only be linked to a point with smaller function value that is within distance r_N (provided that a local search has not been applied unnecessarily, and the starting point is added to the resulting minimum for that reason only). Moreover (under the same provision), if x is assigned to a cluster which is initiated by a local minimum x^*, then there exists an r_N-descent sequence connecting x and x^*. The sample point x can be assigned to several clusters, if there exist r_N-descent sequences from x to each of the corresponding local minima.

Unfortunately, even if r_N tends to 0, then the fact that there exists an r_N-descent sequence from x to a local minimum x^*, does not necessarily imply that $x \in R_{x^*}$. If P is applied to x, then it is still possible that it will follow another descent path, and find another (possibly undetected) local minimum. However, as we will see later, this cannot happen if x is located in the interior of a component which includes some local minimum as its only stationary point.

To understand the advantage of this approach over Single Linkage, let us consider the one dimensional example in Figure 1.



Suppose that $x_1,\ldots,x_5$ are reduced sample points which are ordered according to their function value. Both Single Linkage and the above procedure will start by applying P to x_1. Single Linkage will then assign all points $x_1,\ldots,x_5$ to the cluster which is initiated by the local minimum x^*, thus missing the global minimum x_*. The above procedure will assign x_2 to the cluster which is initiated by x^*. But at the moment that x_3 is considered, it is not possible to link x_3 to x^*, since $\|x_3-x^*\| > r_N$. Thus, P will be applied to x_3 and the global minimum x_* is located.

Intuitively speaking, any two local minima will always be separated by a region with higher function values, so that the above procedure will locate every local minimum in the neighbourhood of which a point has been sampled if r_N is small enough.

Since the function values are used in an explicit way in the clustering process, it is no longer necessary to reduce the sample. Note that it is not even essential to actually assign the sample points to clusters. For every sample point x, the decision whether P should be applied to x does not depend on the cluster structures; the decision only depends on the fact whether or not there exists a sample point z with $f(z) < f(x)$ within distance r_N of x. We now turn to an algorithm in which the superfluous clustering is omitted altogether.

Multi Level Single Linkage [Timmer 1984]

Step 1. For every $i = 1,\ldots,N$ apply P to the sample point x_i except if $x_i \in (S-S_\tau) \cup X^*_\upsilon$ or if there is a sample point x_j with $f(x_j) < f(x_i)$ and $\|x_j-x_i\| \leq r_N$.
Add new local minima encountered during the local search to X^*.

For this method it can be proved [Timmer 1984] that if r_N is chosen according to (21) with $\sigma > 0$, and if x is an arbitrary sample point, then the probability that P is applied to x tends to 0 with increasing N. If $\sigma > 2$, the probability that a local search is applied tends to 0 with increasing N. If $\sigma > 4$, then, even if the sampling continues forever, the total number of local searches ever started is finite with probability 1. Furthermore, if r_N tends to 0, than any local minimum x^* will be found within a finite number of iterations with probability 1.

Obviously, this final asymptotic correctness result justifies application of the stopping rules developed for Multistart to Multi Level Single Linkage. We refer the reader to [Timmer 1984] for a more extensive discussion of the Multi Level Single Linkage method. (Technical reports describing further details will also shortly be available from the authors.)

4. COMPUTATIONAL EXPERIMENTS

In this section we shall discuss the computational performance of the methods described in Sections 3.3 and 3.4 on a number of test problems. For this purpose the algorithms were coded in Fortran IV and run on the DEC 2060 computer of the Computer Institute Woudestein.

To be able to compare our methods with other existing ones, the unconstrained methods have been tested on the standard set of test functions [Dixon & Szegö 1978], which is commonly used in global optimization. Since all test functions are twice continuously differentiable, we used the VA10AD variable metric subroutine from the Harwell Subroutine Library as the local search procedure in all (unconstrained) experiments.

To obtain an impression of the numerical performance of the Single Linkage methods we applied them to four independent samples of size 1000. For all three methods we reduced the sample to 100 points ($q=0.1$) and set σ equal to 4. Furthermore, we chose both υ and τ to be equal to zero in all experiments, thus neglecting the set $S-S_\tau$ and X^*_υ. If, however, a local search was performed resulting in a so far undetected minimum, then we replaced the starting point of the search by the newly detected minimum, to prevent a local search from being started close to this minimum in every succeeding iteration.

The average results of the four runs are listed in Table 1.

Table 1.

Samples of size 1000

		Single Linkage	Multi Level Single Linkage
Function			
GP:	l.m.	3	3
	l.s.	3	3
	f.e.	163	91
BR:	l.m.	3	3
	l.s.	3	3
	f.e.	157	65
H3:	l.m.	2	2
	l.s.	2	4
	f.e.	161	112
H6:	l.m.	2	2
	l.s.	5	10
	f.e.	585	986
S5:	l.m.	5	5
	l.s.	5	5
	f.e.	324	211
S7:	l.m.	6(*)	6(*)
	l.s.	6	6
	f.e.	429	281
S10:	l.m.	7(*)	8(*)
	l.s.	7	8
	f.e.	439	346

l.m.: number of local minima found

l.s.: number of local searches performed

f.e.: number of function evaluations required (not including the 1000 function evaluations required to determine the function values of the sample points)

(*): Global minimum was not found in one of the four runs

In one of the four runs the methods did not find the global minimum of both the S7 and the S10 test function. The reasons for this are twofold. Firstly, the global minimum of these functions is relatively close to other local minima. Secondly, one of the four samples happened to be a very unfortunate one: the regions of attraction surrounding the region of attraction of the global minimum contained sample points whose function values were smaller than the smallest function value attained in a sample point in the region of attraction of the global minimum. (Note that in the case of the S7 test function, the glcbal minimum was the only minimum that was not found).

It is possible to implement the methods such that the global minimum of every test function is found in each of the four runs. For instance, this will be achieved if a steepst descent step is performed from every reduced sample point and the methods are applied to the resulting transformed sample. A small value of σ (e.g. σ=2) will also cause the methods to find the global minimum of every test function in each of the four runs. However, both changes will increase the number of function values required.

The number of local searches started unnecessarily is the largest for the test functions H3 and H7. This is due to the fact that these functions are badly scaled.

The computational experiments are continued with Multi Level Single Linkage. This method has been compared with a few leading contenders whose computational behaviour is described in [Dixon & Szegö 1978]. In this reference methods are compared on the basis of two criteria: the number of function evaluations and the running time required to solve each of the seven test problems. To eliminate the influence of the different computer systems used, the running time required is measured in units of standard time, where one unit corresponds to the running time needed for 1000 evaluations of the S5 test function in the point (4,4,4,4).

Since both the number of function evalutions and the units of standard time required are sensitive to the peculiarities of the sample at hand, the results reported for Multi Level Single Linkage are the average outcome of four independent runs again. As before we chose $\tau = \upsilon = 0$ and $\sigma = 4$ in our implementation of Multi Level Single Linkage. However, we now applied Multi Level Single Linkage to 20% of the sample points (q=0.2) (the reason that we set q equal to 0.1 before was that a major reduction of the sample is necessary for successful application of Single Linkage). Furthermore, it did not seem reasonable to apply Multi Level Single Linkage to samples of fixed size. After an initial sample of size 100, we increased the sample and applied Multi Level Single Linkage iteratively until the expected number of minima was equal to the number of different local minima observed (cf.20).

In Table 3 and Table 4 we summarize the computational results of the methods listed in Table 2 (except for Multi Level Single Linkage, the results are taken from [Dixon & Szegö 1978]).

Table 2.

Methods

A	Trajectory method [Branin & Hoo 1972]
B	Random direction method [Bremmerman 1970]
C	Controlled Random Search [Price 1978]
D	Method proposed in [Törn 1976, 1978] based on concentration of the sample and density clustering
E	Method based on reduction, density clustering and a spline approximation of the distribution function ϕ of f [De Biase & Frontini 1978]
F	Multi Level Single Linkage

Table 3.

Number of function evaluations

Method	GP	BR	H3	H6	S5	S7	S10
A	-	-	-	-	5500	5020	4860
B	300	160	420L	515	375L	405L	336L
C	2500	1800	2400	7600	3800	4900	4400
D	2499	1558	2584	3447	3649	3606	3874
E	378	597	732	807	620	788	1160
F	148	206	197	487	404	432(*)	564

L : the method did not find the global minimum

(*): the global minimum was not found in one of the four runs

Table 4.

Number of units standard time

	Function						
	GP	BR	H3	H6	S5	S7	S10
Method							
A	-	-	-	-	9	8.5	9.5
B	0.7	0.5	2L	3	1.5L	1.5L	2L
C	3	4	8	46	14	20	20
D	4	4	8	16	10	13	15
E	15	14	16	21	23	30	30
F	0.15	0.25	0.5	2	1	1(*)	2

L : the method did not find the global minimum

(*): the global minimum was not found in one of the four runs

As before Multi Level Single Linkage did not find the global minimum of the S7 test function in one of the four runs. Again, this failure could have been prevented by chosing σ to be equal to 2. In that case, the computational results of the method obtained on the test functions GP, BR, H3, H7 and S5 turn out to be comparable to the numbers given in Table 3 and Table 4. However, the number of function evaluations required for the functions S7 and S10 increase by a factor of 2 and 3 respectively. This is due to the fact that all minima of both functions are found in an early stage if σ equals 2. However, the sample must then be increased considerably before our stopping criterion is satisfied.

Since the stopping rules involved in the methods listed in Table 2 are totally different, the comparison between the methods can never be entirely fair: there is always a trade-off between reliability and computational effort that is hard to measure consistently. However, we feel confident that Multi Level Single Linkage is one of the most reliable and efficient methods presently available.

Acknowledgement. This research was partially supported by the Netherlands Foundation for Mathematics SMC with financial aid from the Netherlands Organisation for Advancement of Pure Research (ZWO). The research of the first author was also partially supported by NSF grant no. ECS-831-6224 and by a NATO Senior Scientist Fellowship.

REFERENCES

Allgower, E. and K. Georg (1980), Simplicial and continuation methods for approximating fixed points and solutions of systems of equations. Siam Review 22, 28-84.

Anderssen, R.S. (1972), Global optimization, in R.S. Anderssen, L.S. Jennings and D.M. Ryan (eds.). Optimization (University of Queensland Press).

Beale, E.M.L. and J.J.H. Forrest (1978), Global optimization as an extension of integer programming. In [Dixon & Szegö 1978].

Becker, R.W. and G.V. Lago (1970), A global optimization algorithm. In Proceedings of the 8th Allerton Conference on Circuits and Systems Theory.

Boender, C.G.E., A.H.G. Rinnooy Kan, L. Stougie and G.T. Timmer (1982), A stochastic method for global optimization. Mathematical Programming 22, 125-140.

Boender, C.G.E. (1984), The Generalized Multinomial Distribution: A Bayesian Analysis and Applications. Ph.D. Dissertation, Erasmus Universiteit Rotterdam (Centrum voor Wiskunde en Informatica, Amsterdam).

Branin, F.H. (1972), Widely convergent methods for finding multiple solutions of simultaneous nonlinear equations. IBM Journal of Research Developments, 504-522.

Branin, F.H. and S.K. Hoo (1972), A method for finding multiple extrema of a function of n variables. In F.A. Lootsma (ed.), Numerical Methods of Nonlinear Optimization (Academic Press, London).

Bremmerman, H. (1970), A method of unconstrained global optimization. Mathematical Biosciences 9, 1-15.

Brooks, S.H. (1958), A discussion of random methods for seeking maxima. Operations Research 6, 244-251.

De Biase, L. and F. Frontini (1978), A stochastic method for global optimization: its structure and numerical performance. In [Dixon & Szegö 1978].

Devroye, L. (1978), Progressive global random search of continuous functions. Mathematical Programming 15, 330-342.

Dixon, L.C.W., J. Gomulka and G.P. Szegö (1975), Towards global optimization. In [Dixon & Szegö 1975].

Dixon, L.C.W. (1978), Global optima without convexity. Technical Report, Numerical Optimisation Centre Hatfield Polytechnic, Hatfield, England.

Dixon, L.C.W. and G.P. Szegö (eds.) (1978), Towards Global Optimization 2 (North-Holland, Amsterdam).

Evtushenko, Y.P. (1971), Numerical methods for finding global extrema of a nonuniform mesh. U.S.S.R. Computing Machines and Mathematical Physics 11, 1390-1404.

Falk, J.E. and R.M. Solund (1969), An algorithm for separable nonconvex programming. Management Science 15, 550-569.

Goldstein, A.A. and J.F. Price (1971), On descent from local minima. Mathematics of Computation 25, 569-574.

Hansen, E. (1980), Global optimization using interval analysis - the multidimensional case. Numerische Mathematik 34, 247-270.

Kuhn, H.W., Z. Wang and S. Xu (1984), On the cost of computing roots of polynomials. Mathematical Programming 28, 156-164.

Levy, A. and S. Gomez (1980), The tunneling algorithm for the global optimization problem of constrained functions. Technical Report, Universidad National Autonoma de Mexico.

McCormick, G.P. (1976), Compatibility of global solutions to factorable nonconvex programming, Part 1 - convex underestimating problems. Mathematical Programming 10, 147-175.

Price, W.L. (1978), A controlled random search procedure for global optimization. In [Dixon & Szegö 1978a].

Rinnooy Kan, A.H.G. and G.T. Timmer (1984), Stochastic methods for global optimization. To appear in the American Journal of Mathematical and Management Sciences.

Rubinstein, R.Y. (1981), Simulation and the Monte Carlo Method (John Wiley & Sons, New York).

Shubert, B.O. (1972), A sequential method seeking the global maximum of a function. Siam Journal on Numerical Analysis 9, 379-388.

Solis, F.J. and R.J.E. Wets (1981), Minimization by random search techniques. Mathematics of Operations Research 6, 19-30.

Solund, R.M. (1971), An algorithm for separable nonconvex programming problems 2. Management Science 17, 759-773.

Timmer, G.T. (1984), Global Optimization: A Stochastic Approach. Ph.D. Dissertation, Erasmus Universiteit Rotterdam (Centrum voor Wiskunde en Informatica, Amsterdam).

Todd, M.J. (1976), The Computation of Fixed Points and Applications Springer Verlag, Berlin).

Törn, A.A. (1976), Cluster analysis using seed points and density determined hyperspheres with an application to global optimization. In Proceeding of the third International Conference on Pattern Recognition, Coronado, California.

Törn, A.A. (1978), A search clustering appraoch to global optimization. In [Dixon & Szegö, 1978].

Treccani, G. (1975), On the convergence of Branin's method: a counter example. In [Dixon & Szegö 1975].

Zielinski, R. (1981), A stochastic estimate of the structure of multi-extremal problems. Mathematical Programming 21, 348-356.

Bar Code
On Last
Page